Analyse Convexe, 4M057
Résumé du Chapitre 5 : *Programmation linéaire*

Sorbonne Université, Master 1

Mars 2020

Page Web du cours :
`https://webusers.imj-prg.fr/~stanislaw.szarek/2020/AC/`

Le but de ce résumé et le but du Chapitre 5

Le but de ce résumé est de donner un aperçu de haut niveau du chapitre pour faciliter une étude individuelle. Les arguments seront informels ; pour des énoncés et des définitions précis, et pour les détails de preuves il faut s'adresser au poly.

Le but de ce résumé et le but du Chapitre 5

Le but de ce résumé est de donner un aperçu de haut niveau du chapitre pour faciliter une étude individuelle. Les arguments seront informels ; pour des énoncés et des définitions précis, et pour les détails de preuves il faut s'adresser au poly.

Le but du Chapitre 5 est de mettre en application les résultats des chapitres précédents dans la programmation linéaire. On va se concentrer sur les aspects théoriques, des points plus pratiques seront traités en TD.

La programmation linéaire

Un problème de programmation linéaire est un problème d'optimisation du type :

$$(\mathcal{P}) \begin{cases} \text{déterminer} & \inf f(x) \\ \text{sous les contraintes} & f_i(x) \leq b_i, \ i = 1, \ldots, m \end{cases}$$

où f et les f_i sont des formes linéaires sur E.

La programmation linéaire

Un problème de programmation linéaire est un problème d'optimisation du type :

$$(\mathcal{P}) \begin{cases} \text{déterminer} & \inf f(x) \\ \text{sous les contraintes} & f_i(x) \leq b_i, \ i = 1, \ldots, m \end{cases}$$

où f et les f_i sont des formes linéaires sur E. Autrement dit, on cherche à minimiser f sur le convexe polyédral C déterminé par les inéquations $\{f_i(x) \leq b_i \ , \ i = 1, \ldots, m\}$.

La programmation linéaire

Un problème de programmation linéaire est un problème d'optimisation du type :

$$(\mathcal{P})\begin{cases} \text{déterminer} & \inf f(x) \\ \text{sous les contraintes} & f_i(x) \leq b_i,\ i = 1, \ldots, m \end{cases}$$

où f et les f_i sont des formes linéaires sur E. Autrement dit, on cherche à minimiser f sur le convexe polyédral C déterminé par les inéquations $\{f_i(x) \leq b_i\ ,\ i = 1, \ldots, m\}$. Géométriquement, on cherche d'un hyperplan d'appui $H = f^{-1}(a)$ de C tel que C est contenu dans le demi-espace D_- correspondant.

La programmation linéaire

Un problème de programmation linéaire est un problème d'optimisation du type :

$$(\mathcal{P})\begin{cases} \text{déterminer} & \inf f(x) \\ \text{sous les contraintes} & f_i(x) \leq b_i,\ i = 1, \ldots, m \end{cases}$$

où f et les f_i sont des formes linéaires sur E. Autrement dit, on cherche à minimiser f sur le convexe polyédral C déterminé par les inéquations $\{f_i(x) \leq b_i\ ,\ i = 1, \ldots, m\}$. Géométriquement, on cherche d'un hyperplan d'appui $H = f^{-1}(a)$ de C tel que C est contenu dans le demi-espace D_- correspondant.

Rappelons que le fait que on a des inégalités du type $\leq$ en place de $\geq$ n'a pas d'importance, on peut toujours remplacer f_i par $-f_i$, b_i par $-b_i$, et $\leq$ par $\geq$. On suppose que $C \neq \emptyset$ et typiquement on va avoir $E = \mathbb{R}^p$.

La programmation linéaire

Un problème de programmation linéaire est un problème d'optimisation du type :

$$(\mathcal{P})\begin{cases} \text{déterminer} & \inf f(x) \\ \text{sous les contraintes} & f_i(x) \leq b_i,\ i = 1, \ldots, m \end{cases}$$

où f et les f_i sont des formes linéaires sur E. Autrement dit, on cherche à minimiser f sur le convexe polyédral C déterminé par les inéquations $\{f_i(x) \leq b_i\ ,\ i = 1, \ldots, m\}$. Géométriquement, on cherche d'un hyperplan d'appui $H = f^{-1}(a)$ de C tel que C est contenu dans le demi-espace D_- correspondant.

Rappelons que le fait que on a des inégalités du type $\leq$ en place de $\geq$ n'a pas d'importance, on peut toujours remplacer f_i par $-f_i$, b_i par $-b_i$, et $\leq$ par $\geq$. On suppose que $C \neq \emptyset$ et typiquement on va avoir $E = \mathbb{R}^p$.

Les éléments de C sont appelés les programmes admissibles de $(\mathcal{P})$ et une solution de $(\mathcal{P})$, c.a.d., $x_0 \in C$ tel que $f(x_0) = \inf_{x \in C} f(x)$ est appelé un programme optimal pour $(\mathcal{P})$.

Une reduction au cas “non-trivial”

On commence par se débarrasser des cas “triviaux” quand $\inf_{x \in C} f(x) = -\infty$. Ça peut arriver dans deux cas de figure.

Une reduction au cas “non-trivial”

On commence par se débarrasser des cas “triviaux” quand $\inf_{x \in C} f(x) = -\infty$. Ça peut arriver dans deux cas de figure.

1° C contient une droite affine $x + \mathbb{R}u$ (ou, qui est équivalent, $\mathbb{R}u \subset N(C_{as})$) telle que $f(u) \neq 0$; dans ce cas $f(x + tu) = f(x) + tf(u)$ et évidemment $f(x) + tf(u) \to -\infty$ quand $t \to +\infty$ ou $t \to -\infty$ (lequel de deux depend de la signe de $f(u)$).

Une reduction au cas “non-trivial”

On commence par se débarrasser des cas “triviaux” quand $\inf_{x \in C} f(x) = -\infty$. Ça peut arriver dans deux cas de figure.

1° C contient une droite affine $x + \mathbb{R}u$ (ou, qui est équivalent, $\mathbb{R}u \subset N(C_{as})$) telle que $f(u) \neq 0$; dans ce cas $f(x + tu) = f(x) + tf(u)$ et évidemment $f(x) + tf(u) \to -\infty$ quand $t \to +\infty$ ou $t \to -\infty$ (lequel de deux depend de la signe de $f(u)$).

2° C contient une demi-droite affine $x + \gamma_u = x + \mathbb{R}_+ u$ (ou, qui est équivalent, $\gamma_u \subset C_{as}$) telle que $f(u) < 0$; même argument.

Une reduction au cas “non-trivial”

On commence par se débarrasser des cas “triviaux” quand $\inf_{x \in C} f(x) = -\infty$. Ça peut arriver dans deux cas de figure.

1° C contient une droite affine $x + \mathbb{R}u$ (ou, qui est équivalent, $\mathbb{R}u \subset N(C_{as})$) telle que $f(u) \neq 0$; dans ce cas $f(x + tu) = f(x) + tf(u)$ et évidemment $f(x) + tf(u) \to -\infty$ quand $t \to +\infty$ ou $t \to -\infty$ (lequel de deux depend de la signe de $f(u)$).

2° C contient une demi-droite affine $x + \gamma_u = x + \mathbb{R}_+ u$ (ou, qui est équivalent, $\gamma_u \subset C_{as}$) telle que $f(u) < 0$; même argument.

Si $N(C) = N(C_{as}) \neq \{0\}$, mais $f(u) = 0$ pour tout $u \in N(C)$, il suffit de résoudre le problème d'optimisation $(\mathcal{P})$ avec C_0 donné par Proposition 3.7.10 puisque la forme f prend la même valeur $f(x)$ sur le sous-espace affine $c + N(C)$. Autrement dit, on peut se ramener au cas où le cône asymptotique de l'ensemble de programmes admissibles est saillant.

Le cas “non-trivial”

Pour résumer, on s’est ramené au cas quand dans la représentation de C

$$(*) \qquad C = \mathrm{conv}(\mathrm{Ext}\, C) + C_{as} = \mathrm{conv}(\mathrm{Ext}\, C) + \mathrm{cone}(u_1, \ldots, u_s)$$

le cône asymptotique C_{as} est saillant avec les génératrices extrémales $\gamma_{u_1}, \ldots, \gamma_{u_s}$ et $f(u_i) \geq 0$ pour tout $i = 1, \ldots, s$.

Le cas “non-trivial”

Pour résumer, on s’est ramené au cas quand dans la représentation de C

$$(*) \qquad C = \mathrm{conv}(\mathrm{Ext}\, C) + C_{as} = \mathrm{conv}(\mathrm{Ext}\, C) + \mathrm{cone}(u_1, \ldots, u_s)$$

le cône asymptotique C_{as} est saillant avec les génératrices extrémales $\gamma_{u_1}, \ldots, \gamma_{u_s}$ et $f(u_i) \geq 0$ pour tout $i = 1, \ldots, s$.

Avec ces hypothèses, on voit immédiatement que $\inf_{x \in C} f(x)$ est le même que $\inf_{x \in \mathrm{conv}(\mathrm{Ext}\, C)} f(x)$ et alors le même que $\inf_{x \in \mathrm{Ext}\, C} f(x)$, et il est évidemment atteint parce que $\mathrm{Ext}\, C$ est fini.

Le cas “non-trivial”

Pour résumer, on s'est ramené au cas quand dans la représentation de C

$$(*) \qquad C = \mathrm{conv}(\mathrm{Ext}\, C) + C_{as} = \mathrm{conv}(\mathrm{Ext}\, C) + \mathrm{cone}(u_1, \ldots, u_s)$$

le cône asymptotique C_{as} est saillant avec les génératrices extrémales $\gamma_{u_1}, \ldots, \gamma_{u_s}$ et $f(u_i) \geq 0$ pour tout $i = 1, \ldots, s$.

Avec ces hypothèses, on voit immédiatement que $\inf_{x \in C} f(x)$ est le même que $\inf_{x \in \mathrm{conv}(\mathrm{Ext}\, C)} f(x)$ et alors le même que $\inf_{x \in \mathrm{Ext}\, C} f(x)$, et il est évidemment atteint parce que $\mathrm{Ext}\, C$ est fini.

Notons également que la même conclusion tient quand C_{as} n'est pas saillant, sous réserve qu'on ne se trouve pas dans le cas de figure 1° ou 2°, c.a.d., que $\inf_{x \in C} f(x) > -\infty$.

Résolution "théorique" de $(\mathcal{P})$

En vue de considérations sur la page précédente, une manière de résoudre le problème $(\mathcal{P})$ (sous l'hypothèse que C ne contient pas d'une droite affine, ou $N(C) = \{0\}$) consiste donc à :

- Déterminer l'ensemble fini $\mathrm{Gext}(C_{as})$.
- Tester si il existe $\gamma_u \in \mathrm{Gext}(C_{as})$ tel que $f(u) < 0$; si oui, $\inf_{x \in C} f(x) = -\infty$.
- Dans le cas contraire déterminer l'ensemble fini $\mathrm{Ext}\, C$.
- Calculer $f(x)$ pour tous les $x \in \mathrm{Ext}\, C$, et leur minimum.

Résolution "théorique" de $(\mathcal{P})$

En vue de considérations sur la page précédente, une manière de résoudre le problème $(\mathcal{P})$ (sous l'hypothèse que C ne contient pas d'une droite affine, ou $N(C) = \{0\}$) consiste donc à :

- Déterminer l'ensemble fini $\mathrm{Gext}(C_{as})$.
- Tester si il existe $\gamma_u \in \mathrm{Gext}(C_{as})$ tel que $f(u) < 0$; si oui, $\inf_{x \in C} f(x) = -\infty$.
- Dans le cas contraire déterminer l'ensemble fini $\mathrm{Ext}\, C$.
- Calculer $f(x)$ pour tous les $x \in \mathrm{Ext}\, C$, et leur minimum.

Tous ces étapes sont en principe faisable car on a montré que trouver $\gamma_u \in \mathrm{Gext}(C_{as})$ et $x \in \mathrm{Ext}\, C$ dépend de résolution d'un système d'équations linéaires, que on peut faire efficacement par l'élimination de Gauss-Jordan.

Résolution "théorique" de $(\mathcal{P})$

En vue de considérations sur la page précédente, une manière de résoudre le problème $(\mathcal{P})$ (sous l'hypothèse que C ne contient pas d'une droite affine, ou $N(C) = \{0\}$) consiste donc à :

- Déterminer l'ensemble fini $\mathrm{Gext}(C_{as})$.
- Tester si il existe $\gamma_u \in \mathrm{Gext}(C_{as})$ tel que $f(u) < 0$; si oui, $\inf_{x \in C} f(x) = -\infty$.
- Dans le cas contraire déterminer l'ensemble fini $\mathrm{Ext}\, C$.
- Calculer $f(x)$ pour tous les $x \in \mathrm{Ext}\, C$, et leur minimum.

Tous ces étapes sont en principe faisable car on a montré que trouver $\gamma_u \in \mathrm{Gext}(C_{as})$ et $x \in \mathrm{Ext}\, C$ dépend de résolution d'un système d'équations linéaires, que on peut faire efficacement par l'élimination de Gauss-Jordan. Le premier étape, détermination de $N(C)$, dépend également de résolution d'un système d'équations linéaires.

Les difficultés

Pourquoi est-ce une solution "théorique" plutôt que pratique ? La "seule" difficulté est que le nombre de systèmes d'équations linéaires à résoudre peut être énorme si p, N sont grands : les bornes étant respectivement C_N^p et C_N^{p-1}. Dans l'example suggéré dans le chapitre 4 ($p = 3, N = 4$) on a eu 4 systèmes à résoudre pour déterminer $\mathrm{Ext}\, C$ et 6 systèmes pour $\mathrm{Gext}(C_{as})$, que on a pu faire manuellement. Mais pour, disons, $p = 50$ et $N = 100$ (pas vraiment très grands pour des application réelles), le nombre de systèmes est plus grand que 10^{29}, impossible même en utilisant un super-ordinateur.

Les difficultés

Pourquoi est-ce une solution "théorique" plutôt que pratique ? La "seule" difficulté est que le nombre de systèmes d'équations linéaires à résoudre peut être énorme si p, N sont grands : les bornes étant respectivement C_N^p et C_N^{p-1}. Dans l'example suggéré dans le chapitre 4 ($p = 3, N = 4$) on a eu 4 systèmes à résoudre pour déterminer $\mathrm{Ext}\, C$ et 6 systèmes pour $\mathrm{Gext}(C_{as})$, que on a pu faire manuellement. Mais pour, disons, $p = 50$ et $N = 100$ (pas vraiment très grands pour des application réelles), le nombre de systèmes est plus grand que 10^{29}, impossible même en utilisant un super-ordinateur.

Il faut alors utiliser des astuces en place de la force brutale. Il existe des approches qui marchent toujours assez bien, par exemple l'algorithme de points intérieurs. Une autre méthode, l'algorithme du simplexe, est très efficace en pratique, même que on peut montrer que il peut échouer pour certains problèmes assez particulieurs.

Résolution “pratique” : les astuces

Le nombre de systèmes d’équations à résoudre étant potentiellement très grand, il est important de les bien cataloguer (ou paramétriser). À cette fin, il est commode de se limiter aux problèmes des formes particulières dites “canoniques” ou “standard” (mais pas “trop” particulières puisque on veut pouvoir réduire un problème quelconque à un problème canoniques ou standard), peut-être en dimension un petit peu plus grande.

Résolution “pratique” : les astuces

Le nombre de systèmes d'équations à résoudre étant potentiellement très grand, il est important de les bien cataloguer (ou paramétriser). À cette fin, il est commode de se limiter aux problèmes des formes particulières dites “canoniques” ou “standard” (mais pas “trop” particulières puisque on veut pouvoir réduire un problème quelconque à un problème canoniques ou standard), peut-être en dimension un petit peu plus grande.

En dehors de questions de paramétrisation, qui sont assez techniques, la stratégie la plus naturelle de recherche d'un programme optimal est basée sur une procédure itérative. Les grandes lignes de cette procédure sont expliquer sur la page suivante ; un exemple sera présenté en TD.

Résolution “pratique” : une stratégie itérative

• Pour initialisation, on trouve x_0, un sommet (c.a.d., un point extremal) quelconque de C

Résolution “pratique” : une stratégie itérative

• Pour initialisation, on trouve x_0, un sommet (c.a.d., un point extremal) quelconque de C

• on examine tous les arêtes (faces de dimension 1) issue de x_0 ; il y a plusieurs cas de figure :
– s'il existe une arête infinie $x_0 + \gamma_u$ telle que $f(u) < 0$, il suit que $\inf_{x \in C} f(x) = -\infty$
– sinon, et si pour toute arête finie $[x_0, y]$ on a $f(y) \geq f(x_0)$, il suit que x_0 est un programme optimal
– s'il existe une arête finie $[x_0, y]$ telle que $f(y) < f(x_0)$, on pose $x_1 = y$ (on note que forcement $x_1 \in \mathrm{Ext}\, C$)

Résolution “pratique” : une stratégie itérative

• Pour initialisation, on trouve x_0, un sommet (c.a.d., un point extremal) quelconque de C

• on examine tous les arêtes (faces de dimension 1) issue de x_0 ; il y a plusieurs cas de figure :
– s'il existe une arête infinie $x_0 + \gamma_u$ telle que $f(u) < 0$, il suit que $\inf_{x \in C} f(x) = -\infty$
– sinon, et si pour toute arête finie $[x_0, y]$ on a $f(y) \geq f(x_0)$, il suit que x_0 est un programme optimal
– s'il existe une arête finie $[x_0, y]$ telle que $f(y) < f(x_0)$, on pose $x_1 = y$ (on note que forcement $x_1 \in \mathrm{Ext}\, C$)

• on continue en remplaçant x_0 par x_1

La stratégie itérative : la conclusion

Etant donné que $\mathrm{Ext}\, C$ est fini, cette procédure doit s'arrêter : soit on établit que $\inf_{x \in C} f(x) = -\infty$, soit on trouve un programme optimal. On espère que le nombre d'itérations ne sera pas trop grand.

La stratégie itérative : la conclusion

Etant donné que $\mathrm{Ext}\, C$ est fini, cette procédure doit s'arrêter : soit on établit que $\inf_{x \in C} f(x) = -\infty$, soit on trouve un programme optimal. On espère que le nombre d'itérations ne sera pas trop grand.

Mais en supposant qu'on veut être efficace, il y a des plusieurs subtilités qui sont évidentes.

La stratégie itérative : la conclusion

Etant donné que $\mathrm{Ext}\, C$ est fini, cette procédure doit s'arrêter : soit on établit que $\inf_{x \in C} f(x) = -\infty$, soit on trouve un programme optimal. On espère que le nombre d'itérations ne sera pas trop grand.

Mais en supposant qu'on veut être efficace, il y a des plusieurs subtilités qui sont évidentes. Par exemple, pour un problème général, même la initialisation (c.a.d., trouver un sommet quelconque de C) n'est pas absolument triviale : il peut facilement arriver qu'une solution du système d'équations linéaires choisi nous donne un point, qui n'appartient pas à C.

La stratégie itérative : la conclusion

Etant donné que $\mathrm{Ext}\, C$ est fini, cette procédure doit s'arrêter : soit on établit que $\inf_{x \in C} f(x) = -\infty$, soit on trouve un programme optimal. On espère que le nombre d'itérations ne sera pas trop grand.

Mais en supposant qu'on veut être efficace, il y a des plusieurs subtilités qui sont évidentes. Par exemple, pour un problème général, même la initialisation (c.a.d., trouver un sommet quelconque de C) n'est pas absolument triviale : il peut facilement arriver qu'une solution du système d'équations linéaires choisi nous donne un point, qui n'appartient pas à C. Puis, à chaque étape on peut avoir plusieurs arêtes $[x_k, y]$ avec $f(y) < f(x_k)$, et il faut établir une règle pour choisir x_{k+1}.

Application de la programmation linéaire : l'acquisition comprimée

On considère un vecteur inconnu $x \in E = \mathbb{R}^n$ que on veut trouver (ou, au moins, bien approximer) en effectuant des mesures linéaires : $f_1(x), \ldots, f_m(x)$, où $f_i \in E'$.

Application de la programmation linéaire : l'acquisition comprimée

On considère un vecteur inconnu $x \in E = \mathbb{R}^n$ que on veut trouver (ou, au moins, bien approximer) en effectuant des mesures linéaires : $f_1(x), \ldots, f_m(x)$, où $f_i \in E'$.

Dans le cas général, on a besoin n mesures pour récupérer x exactement et on peut prendre comme (f_i) n'importe quelle base de E'. Cependant, ce nombre peut être considérablement réduit si nous savons *a priori* que le vecteur x possède une structure particulière, par exemple s'il est parcimonieux (ou éparse), c'est-à-dire que la plupart de coordonnées de x sont nulles.

Certains aspects de ce phénomène spectaculaire et inattendu, y compris une indication du rôle joué par la programmation linéaire, seront esquissés en TD.